

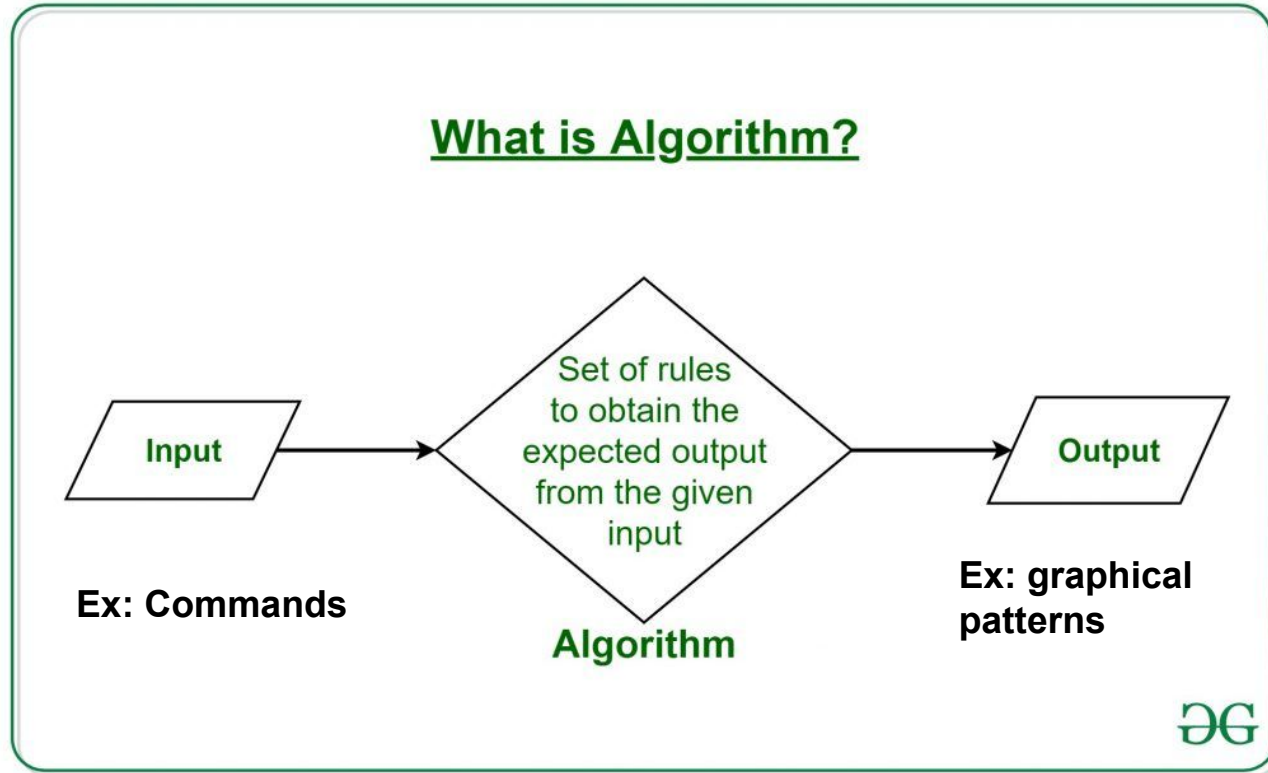
Class 1: Introduction to Robotics

Computer Science & Math

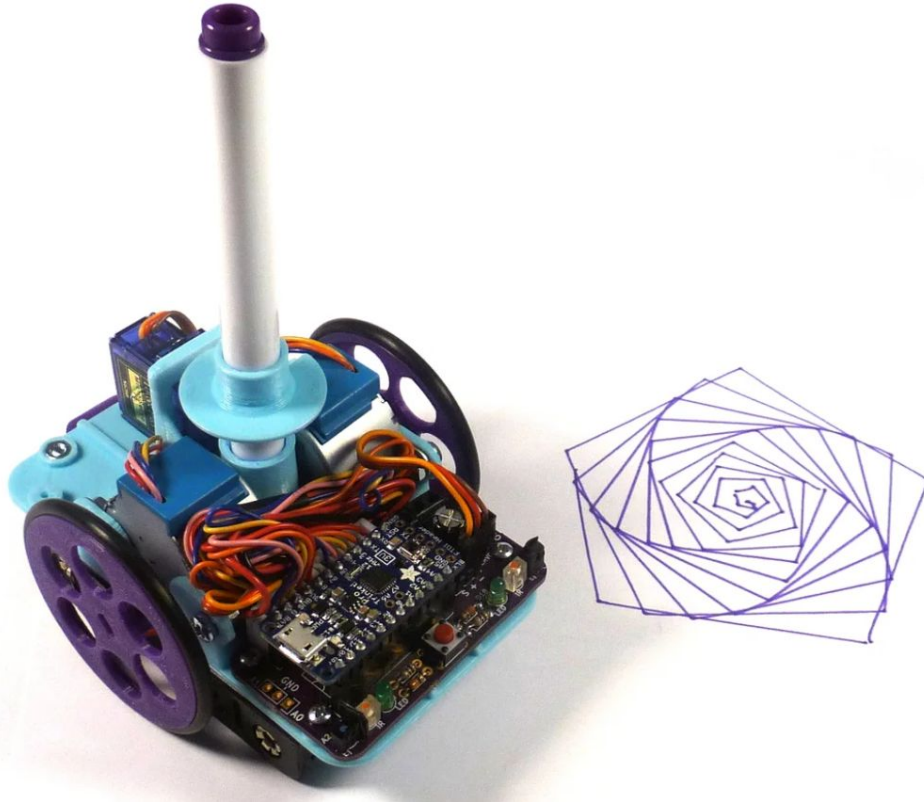
Electrical Engineering

Mechanical Engineering

Computer Science & Math: Concept: **Algorithm**



Our Tool: Drawing Turtle Robot



Turtle Commands:

`forward(mm)`

`backward(mm)`

`left(degrees)`

`right(degrees)`

`penup()`

`pendown()`

Hands-on: Algorithm : Draw a Square

Test your idea in simulation: <https://skulpt.org/>

Run the sample code below. What do you see?

```
from turtle import *  
  
pendown()  
  
for i in range(4):  
    forward(75)  
    left(90)  
  
penup()
```

Note: the turtle starts facing right, or X+ direction

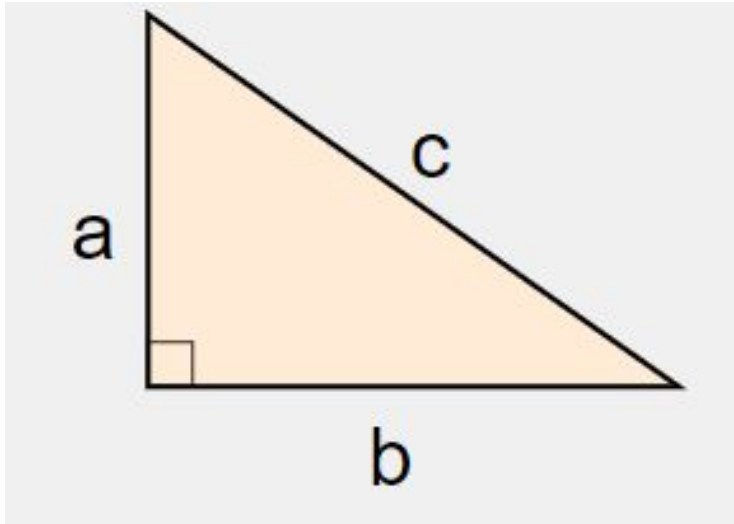


Note: angle is measured counter-clockwise or CCW

Hands-on: Algorithm : Draw a Right Triangle

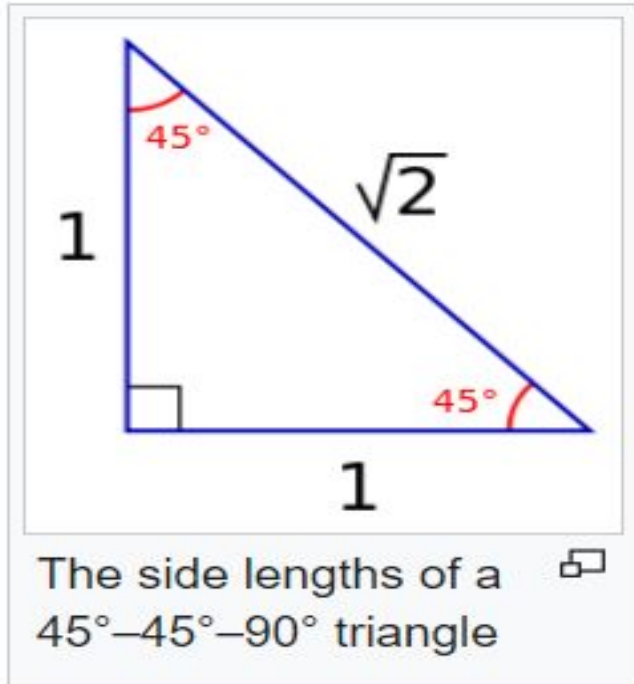
Hint #1: Recall **Pythagoras theorem** (570 BC)

Start with simple triangles with known angles, like 45-45-90 or 30-60-90 triangle



$$c = \sqrt{a^2 + b^2}$$

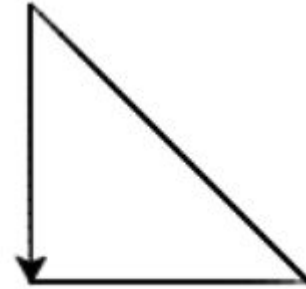
Hands-on: Algorithm : Draw a Right Triangle



1. Get the ratios of the 3 sides
2. Note the 3 known angles
3. Scale up the sides to mm

Hands-on: Algorithm : Draw a Right Triangle

```
from turtle import *  
  
pendown()  
  
forward(100) # move 100mm, scale x100  
left(135) # turn 90+45 deg CCW  
forward(141) #  $\text{sqr-root}(2) \times 100$   
left(135) # another 90+45 deg CCW  
forward(100)  
  
penup()
```



in codes means
“comments for human”, not
executed by the computer

Hands-on: Algorithm : Repeat (once you see the pattern)

Recall how we draw the square earlier?

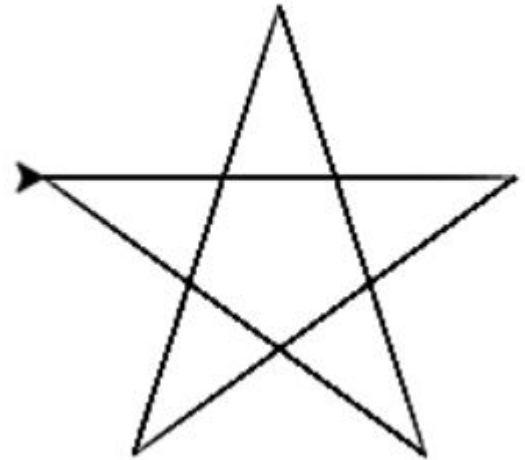
```
for x in range(4): # repeat the code block below 4 times
```

```
    forward(75)  
    left(90)
```

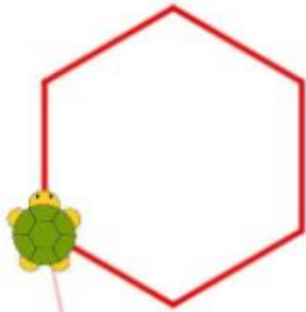


Hands-on: Algorithm : Repeat (once you see the pattern)

```
from turtle import *  
  
pendown()  
  
for x in range(5):  
    forward (170)  
    right(144)  
  
penup()
```



Hands-on: Algorithm : Repeat (once you see the pattern)



Questions:

Can you draw any other shapes with the Turtle commands?

If you know the length of the sides, how do you figure out the angle? And vice versa?

Hint: Sine, Cosine and Tangent functions

<https://www.mathsisfun.com/sine-cosine-tangent.html>

Common problems:

Note the Python turtle library inside the robot is only a SUBSET of the original. Thus NOT all commands in simulation will work in the robot. Use only those given in this presentation.

1. NO negative numbers in forward()
2. NO back(), use backward()